Listing of Claims

1. (Currently Amended) A <u>computer-implemented</u> method for authoring a speech application, comprising the steps of:

creating one or more reusable VoiceXML dialog components;

creating an associated parameter object for each of the reusable VoiceXML dialog components; and

creating a VoiceXML document comprising code for invoking a reusable VoiceXML dialog component and code for configuring the invoked reusable VoiceXML dialog component using an associated parameter object,

wherein the step of creating a reusable VoiceXML dialog component comprises creating a re-entrant reusable VoiceXML dialog component that allow reusable VoiceXML dialog components to be one of initiated, interrupted, inspected, or resumed with a partially filled result object or state object.

2. (Original) The method of claim 1, further comprising the step of populating each associated parameter object with appropriate parameter values.

3. (Original) The method of claim 1, wherein the code for invoking a reusable VoiceXML dialog component comprises a subdialog element.

4. (Original) The method of claim 1, wherein the parameter object comprises a ECMAScript parameter object.

5. (Original) The method of claim 1, further comprising the step of building a library of reusable VoiceXML documents.

6. (Original) The method of claim 1, comprising the step of building a reusable VoiceXML dialog module comprising a standardized set of reusable VoiceXML dialog components.

2

7. (Original) The method of claim 1, wherein the parameter object comprises one of default prompts, object-specific resources, constructors that combine default and application specific parameters, methods for manipulating parameter content, and a combination thereof.

8. (Original) The method of claim 1, wherein the VoiceXML document further comprises code for dynamically compiling a grammar.

9. (Canceled)

10. (Previously Presented) The method of claim 1, wherein re-entrant objects are used for mixed initiative.

11. (Original) The method of claim 1, wherein the VoiceXML document comprises code for calling application-specific objects comprising interaction objects and service objects.

12. (Original) The method of claim 11, wherein the code for calling an interaction object comprises a subdialog element and wherein the code for calling a service object comprises an object element.

13. (Previously Presented) A speech application server, comprising:

a VoiceXML processor for parsing and rendering a VoiceXML document; and

a library comprising one or more reusable VoiceXML dialog components that are accessible by the VoiceXML processor, wherein the VoiceXML document comprises code for invoking a reusable VoiceXML dialog component and code for configuring the invoked reusable VoiceXML dialog component using an associated parameter object,

wherein the reusable VoiceXML dialog components comprise one or more re-entrant reusable VoiceXML dialog component that allow reusable VoiceXML dialog components to be one of initiated, interrupted, inspected, or resumed with a partially filled result object or state object.

14. (Original) The speech application server of claim 13, wherein a reusable VoiceXML dialog component is invoked using a subdialog element.

15. (Original) The speech application server of claim 13, wherein the parameter object comprises an ECMAScript parameter object.

16. (Original) The speech application server of claim 15, wherein an ECMAScript parameter object comprises a container that provides one of default prompts, object specific resources, constructors that combine default and application specific parameters, methods for manipulating parameter content and a combination thereof.

17. (Original) The speech application server of claim 13, wherein the library of reusable VoiceXML dialog components is maintained on a server repository for dynamic access at execution of a reusable VoiceXML dialog component, or maintained on a local repository, or both.

18. (Original) The speech application server of claim 17, wherein the repository further comprises default grammars and audio prompts to support behavior of the reusable VoiceXML dialog components.

19. (Original) The speech application server of claim 17, wherein the library further maintains a reusable VoiceXML dialog module comprising a standardized set of reusable VoiceXML dialog components.

20. (Original) The speech application of claim 19, wherein the reusable VoiceXML module supports dialog localization for other languages.

21. (Original) The speech application server of claim 13, further comprising repository of reusable ECMAScript functions.

22. (Original) The speech application server of claim 13, further comprising a repository for dynamic grammar compilers and audio prompt editors, which can be ported to the VoiceXML processor platform.

23. (Original) The speech application server of claim 13, wherein the VoiceXML processor comprises a VoiceXML browser.

24. (Original) The speech application server of claim 13, wherein the speech application server provides a speech interface for a multi-modal browser.

25. (Canceled)

26. (Previously Presented) The speech application server of claim 13, wherein re-entrant objects are used for mixed initiative.

27. (Original) The speech application server of claim 13, wherein a reusable VoiceXML dialog component comprises an object element for providing dynamic data access.

28. (Currently Amended) A computer-implemented method for implementing a speech application, comprising the steps of:
receiving and parsing a VoiceXML document;
invoking a reusable VoiceXML dialog components using a subdialog element;
instantiating an associated parameter object for configuring the invoked reusable VoiceXML document; and
dynamically compiling a grammar for the invoked reusable VoiceXML dialog component.

29. (Original) The method of claim 28, wherein the step of instantiating an associated parameter object comprises using ECMAScript.

30. (Original) The method of claim 28, further comprising the step of maintaining a repository comprising a library of reusable VoiceXML dialog component and associated parameter objects.

31. (Original) The method of claim 29, further comprising maintaining a repository of default grammars and audio prompts.

32. (Previously Presented) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform method steps for implementing a speech application, the method steps comprising:

receiving and parsing a VoiceXML document;

invoking a reusable VoiceXML dialog components using a subdialog element;

instantiating an associated parameter object for configuring the invoked reusable VoiceXML document; and

dynamically compiling a grammar for the invoked reusable VoiceXML dialog component.

33. (Currently Amended) A server-side speech application server, comprising:

a VoiceXML page generation engine for dynamically building a VoiceXML page;

a first database comprising one or more server-side reusable VoiceXML dialog components that are accessible by the VoiceXML page generation engine for generating an intermediate VoiceXML page;

a second database comprising backend data that is accessible by the VoiceXML page generator to insert data in the intermediate VoiceXML page to generate a VoiceXML page that is served to a requesting client.

34. (Original) The server of claim 33, wherein the reusable VoiceXML dialog components comprise beans and wherein the VoiceXML page generation engine comprises a JSP (java server pages) engine.

6